Facultad de Ciencias

Análisis y predicción de series temporales provenientes de un sistema SCADA de una planta de fabricación industrial.

(Analysis and forecasting of time series from a SCADA system of an industrial manufacturing plant.)

Trabajo de Fin de Máster
para acceder al

Máster en Ciencia de Datos

Autor: Carlos A. Meneses Agudo
Director: Antonio S. Cofiño González
Co-Director: Diego Tuccillo
2 de julio de 2019

Agradecimientos

En primer lugar, quiero agradecer a Antonio y Diego por la guía en el desempeño de este trabajo. Sobre todo por el apoyo a una formación crítica indispensable en el camino científico.

En segundo lugar, agradecer a la empresa CIC Consulting Informático la posibilidad de haber comenzado este proyecto tan interesante como estimulante. En especial a David Vilasack por su total disposición a apoyarme en el desarrollo del proyecto.

También me gustaría agradecer a todos los responsables y profesores del máster por ofrecernos la oportunidad de cursar un máster tan completo como éste.

Y, por último, deseo agradecer de igual forma a mi familia, amigos y pareja por el apoyo recibido durante este pequeño periodo de mi vida.

Resumen

En la presente memoria se recoge el trabajo realizado sobre series temporales sobre consumos energéticos y de recursos de una planta industrial. Se ha tratado de realizar un trabajo de *data science* completo. Se ha partido de unos datos y se ha elaborado un preprocesado y curado de los datos, así como una analítica de estos, tanto de forma cualitativa como cuantitativa.

En general, este trabajo ejemplifica, aunque de forma comedida, el trabajo de *data scientist* sobre un problema real, en el que debemos recolectar los datos, curarlos y preprocesarlos y obtener un cierto valor añadido sobre estos datos. En nuestro ejemplo, nos hemos centrado en tratar de obtener unas predicciones con un conjunto de modelos seleccionado. Para ello se ha tomado una ventana deslizante temporal de 8 pasos hacia el pasado para predecir un paso hacia el futuro. El principal resultado obtenido lo han conseguido dos modelos lineales: Regresión lineal y SVR con kernel lineal.

Esto resulta destacable a la hora de dar valor a estos modelos lineales que, incluso con su sencillez, en ocasiones se comportan mejor que modelos más adecuados a las características del problema. En nuestro caso se debe a que el resto de modelos complejos no han sido lo suficientemente ajustados y en esas condiciones los modelos lineales han resultado vencedores.

De cualquier forma, el valor final de este trabajo reside en todo el proceso aplicado y las líneas de trabajo futuro abiertas.

Palabras clave: Series temporales, *machine learning*, *data science*, industria.

Abstract

This report includes the research carried out on time series on energy and resource consumption of an industrial plant. An attempt has been made to carry out a complete data science. Some data have been used as a starting point and a preprocessing and curing of the data has been elaborated.

In general, this research exemplifies the work of a data scientist on a real problem, in which we must collect the data, cure and preprocess them and obtain a certain added value on these data. In our example, we have focused on trying to get predictions with a selected set of models. For this, we have taken a time slider window of 8 steps into the past to predict 1 step into the future. The main result obtained has been achieved by two linear models: Linear Regression and SVR with linear kernel.

This is remarkable when it comes to giving value to these linear models which, even with their simplicity, sometimes behave better than models more suited to the characteristics of the problem. In our case it is due to the fact that the rest of the complex models have not been enough adjusted. In these conditions the results of the linear models have been succesful.

In any case, the final value of this research lies in the entire process applied and the future lines of research open.

Keywords: Time series, machine learning, data science, industry.

Índice general

Índice de figuras

1 | Introducción

Una serie temporal es una secuencia de observaciones ordenados cronológicamente sobre una característica (serie univariante o escalar) o sobre varias características (serie multivariante o vectorial) [1, 2, 3, 4]. Los valores de una serie temporal van ligados al instante de tiempo en el que son medidos, de esta manera el análisis de las series temporales tiene un carácter especial, pues implica el manejo conjunto de, al menos, dos variables: La variable que se estudia y la variable de tiempo.

El análisis de las series engloba un conjunto de técnicas que permiten extraer todas las regularidades que se observan en el comportamiento pasado y, así, poder tener "mecanismos" para predecir sus valores en el futuro.

Al igual que lo descrito en [4], la metodología actual para analizar series temporales es la confluencia de varias líneas y aporte de trabajos desarrollados en distintos campos científicos muy heterogéneos (debido a la importancia y presencia que ha tenido este concepto para todos estos ámbitos). En éste sentido y a modo de resumen, el desarrollo de distintas técnica o métodos se pueden identificar históricamente en cinco campos de trabajo principales:

El primero con raíces en el estudio de series climáticas y astronómicas, dando origen a la teoría de procesos estocásticos estacionarios desarrollados por Kolmogorov, Wiener y Cramer en la primera mitad del Siglo XX.

El segundo campo lo constituyen los llamados métodos de alisado, introducidos por investigadores en el desarrollo del control de calidad en procesos industriales operativos procesos operativos para series de producción y venta en los años 60s y 70s.

En el tercero encontraríamos la teoría de predicción y control de sistemas lineales, desarrollada en ingeniería de control también en los 70s, y estimulada por el desarrollo de la ingeniería aeronáutica.

El cuarto es la teoría de procesos no estacionarios y sistemas no lineales, desarrollados por estadísticos, económetras y físicos en los últimos años del siglo XX.

Podemos comprobar que todos los métodos disponibles en la actualidad, así como los que se desarrollan actualmente como nuevos, son el aporte de un amplio grupo de profesionales de distintos ámbitos como son los matemáticos, economistas, estadísticos, ingenieros, físicos, etc. Esto nos da una pequeña noción del peso que ha tenido este concepto que ha aglutinado los esfuerzos de una vasta comunidad científica.

1.1 | Ejemplos de uso de series temporales

En general se pueden encontrar numerosos estudios de carácter científico-técnico en el que el concepto de serie temporal este involucrado. A continuación se

presentan unos pequeños ejemplos de diversas áreas y comentaremos brevemente un par para, así, dejar constancia del poder que tiene controlar las técnicas de análisis y predicción de las series temporales.[5]

- Economía: Precio de acciones, mercado de divisas, consumos energéticos en industria, etc. [6, 7]
- Meteorología: Series temporales de variables como la temperatura, cantidad de lluvia, etc. [8]
- Demografía: Evolución temporal de la población total, tasa de natalidad, etc. [9]
- Medicina: Electrocardiogramas, electroencefalogramas, etc. [10, 11]
- Astronomía: Astrometría, fotometría, etc. [12, 13]
- Industria: Predicción de consumos, detección de anomalías, etc. [14]

A modo de introducción de la relevancia de estos ejemplos se han seleccionado un par de estos ejemplos que se introducirán a continuación.

1.1.1. Clasificación de anomalías cardíacas

El electrocardiograma (ECG) se puede usar de manera confiable como medida para controlar la funcionalidad del sistema cardiovascular. Recientemente, ha habido una gran atención hacia la categorización precisa de los latidos del corazón. El estado de arte actual son métodos basado en redes neuronales convolucionales más redes neuronales recurrentes para la clasificación de los latidos del corazón que es capaz de clasificar con precisión cinco arritmias diferentes de acuerdo con el estándar AAMI EC57 [15]. En la figura 1.1 se puede ver como podemos detectar una isquemia a través del ECG del paciente.

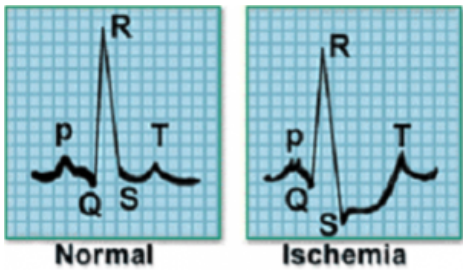


Figura 1.1: Representación esquemática de dos secuencias posibles de un ECG. Fuente: [15].

1.1.2. Diarización de locuciones

En primer lugar, la finalidad de un sistema de diarización de locutores consiste en la separación de una señal de voz en los diferentes locutores que

aparecen en ella. En este campo se está trabajando actualmente con un enfoque de redes neuronales recurrentes. La arquitectura aprende de forma simultánea una incrustación dimensional fija para segmentos acústicos de longitud variable y una función de puntuación para medir la probabilidad de que los segmentos se originaron en el mismo orador o en diferentes oradores [16, 17, 18]. Se puede ver un esquema del objetivo de la diarización en la figura 1.2

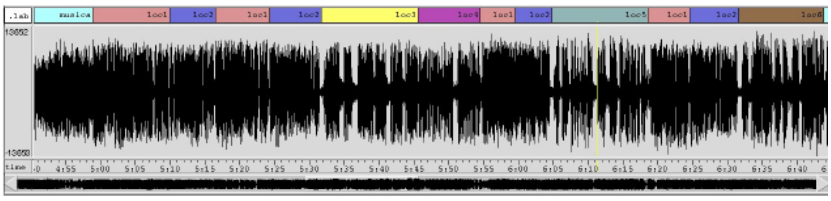


Figura 1.2: Representación de la señal de audio segmentada por locutores. Fuente: [18].

1.2 | Descripción del problema

En España, la ley 54/1997 del sector eléctrico (1997) estableció un proceso gradual para la entrada de los consumidores en el mercado libre de electricidad. Esta ley permite al cliente varias opciones a la hora de contratar electricidad. Por un lado puede acudir a una comercializadora, al mercado mayorista o contratando directamente con un productor. Para un usuario con una gran demanda de electricidad (fábricas, por ejemplo) resulta muy importante llevar un control y una previsión del consumo que se va a realizar en todos los procesos productivos debido a que éstos usuarios deben acudir al mercado mayorista para proveer de energía sus fábricas. De forma resumida, las fábricas acuden a mercados mayoristas para negociar una cierta cantidad de energía a consumir en un determinado plazo de tiempo.

Para poder obtener todo el conocimiento del comportamiento en base a los consumos de una fábrica es necesario la implementación o desarrollo de un sistema de supervisión o monitorización. En concreto, estos procesos son automatizados por sistemas ***S****upervisory* ***C****ontrol* ***A****nd* ***D****ata* ***A****cquisition*, SCADA, es un tipo de de *software* que permite recibir datos operativos a cerca de un sistema físico con el fin de controlar y optimizar los procesos productivos. Sin embargo, la información aportada por estos sistemas suele ser bastante escasa y se reduce a ciertas visualizaciones de las series. En base a esto es posible crear ciertas herramientas de técnicas de supervisión avanzadas que sean capaces de extraer el conocimiento implícito en los datos y trasladarlo a las tomas de decisiones.

La extracción del conocimiento de los datos facilita el análisis de la evolución del consumo, lo cual no es sólo útil en el campo de la eficiencia o negociación de las facturas con la distribuidora, sino también para resolver ciertos problemas en tiempo real en la gestión de desviaciones, planificaciones de consumos y detección de consumos anómalos. En este sentido, nuestra primera gran motivación para emprender este estudio es obtener una predicción en los consumos fiable que

permita usar este conocimiento para incrementar la eficiencia productiva de la fábrica.

1.3 | Objetivos propuestos

Si bien se ha comentado brevemente de forma general todo lo que podemos conseguir al obtener estas predicciones de consumo cabe concretar cuales van a ser nuestros objetivos concretos. En primer lugar, nuestro objetivo primario sigue siendo obtener unas predicciones en un horizonte de 8 horas adecuadas y así poder usarlas en la toma de decisiones de la fábrica.

En paralelo a esto, se debe tener en cuenta todos los objetivos secundarios que se corresponden con apartados técnicos que necesitamos cubrir para obtener nuestro objetivo principal. Todos estos giran en torno a la caracterización y comprensión del conjunto de datos.

Una vez es concluido este primer paso estableceremos una serie de modelos que parten desde dos aproximaciones distintas para obtener una comparativa crítica de cada modelo. Debido al bagaje académico que este máster nos ha aportado intentaremos comparar modelos "clásicos" con modelos propios de *machine learning* y *deep learning* con la idea de establecer todas las ventajas e inconvenientes de ambos mundos.

1.4 | Organización de la tésis

El **Capítulo 1** realizaremos una pequeña introducción del trabajo a un nivel general presentando el concepto de serie temporal y describiendo las motivaciones de nuestra caso de estudio concreto.

El **Capítulo 2** englobará el preprocesado de los datos y el análisis de éstos en cuanto a un análisis de series temporales. Este capítulo alternará las explicaciones teóricas necesarias y su aplicación en los datos.

El **Capítulo 3** consistirá en la explicación de la aproximación clásica a un problema univariado. De la misma forma, este capítulo alternará las explicaciones teóricas oportunas y la aplicación de estos modelos a los datos.

El **Capítulo 4** tratará sobre la aproximación de *machine learning* a este tipo de problemas. De nuevo, se alternarán las explicaciones teóricas sobre los principales algoritmos y su aplicación.

Finalmente, el **Capítulo 5** se dedicará a unas breves conclusiones de lo obtenido así como relatar unas líneas futuras de trabajo en este caso de estudio o similar.

2 | Análisis de los datos

A lo largo de todo este capítulo se va a presentar el conjunto de datos usado así como todas las técnicas realizadas sobre ellos en un punto de vista analítico. A su vez, también se ha trabajado con ciertas representaciones para obtener unas visualizaciones de los datos.

Para comenzar la sección se ve oportuno indicar que la procedencia de los datos es de una fábrica de automóviles y el conjunto de datos procede de un sistema SCADA de uno de los talleres de ésta. Es decir, todas las variables, posteriormente descritas, proceden de un sistema de producción de automóviles.

2.1 | Datos

En primer lugar, se presenta el conjunto de datos. En este caso partimos de 16 variables para el conjunto del *dataset*.

- **Agua potable**: Toda aquella agua destinada al consumo humano. A priori se puede intuir una dependencia de esta variable con el número de trabajadores en la planta en cada instante de tiempo. La variable mide el consumo en m^3 en 15 minutos.
- **Agua permeada**: Tras un proceso de ósmosis inversa obtenemos este agua permeada que consiste en un agua con muy baja salinidad. Por ello esta agua se usa en procesos de pinturas y otros procesos específicos que requieren bajos contenidos en sales. La variable mide el consumo en m^3 en 15 minutos.
- **Agua desionizada**: Este agua consiste en un proceso de osmósis más estricto sobre el agua permeada para obtener un agua con bajo nivel conductivo. De la misma forma este agua se usa en procesos muy específicos que requieren de estas capacidades. La variable mide el consumo en m^3 en 15 minutos.
- **Gas directo**: Es el gas usado para calentar los distintos hornos o baños de cera. Tiene una relación directa con el número de carrocerías producidas. La variable mide el consumo en kWh en 15 minutos.
- **Gas tecnológico**: Es el consumo de energía de agua sobrecalentada pasado a gas por las perdidas de generación de dicha agua sobrecalentada. Se aplica en ciertos procesos industriales y tiene una implicación directa con el numero de carrocerías producido. La variable mide el consumo en kWh en 15 minutos.
- **Gas calefacción**: Es todo el consumo energético para obtener una temperatura ambiente del taller adecuada. La variable mide el consumo en kWh en 15 minutos.

- **Climatizadores**: Energía eléctrica consumida por las turbinas de aire de impulsión de los climatizadores en función de si nos encontramos en un turno productivo o no. En general tiene implicación a la hora de obtener una temperatura agradable dentro del taller. La variable mide el consumo en kWh en 15 minutos.
- **Frío**: Energía eléctrica consumida para producir agua fría que se introduce en los intercambiadores de los climatizadores para lograr la temperatura deseada dentro del taller. La variable mide el consumo en kWh en 15 minutos.
- **Alumbrado**: Energía eléctrica consumida en el alumbrado interior del taller. La variable mide el consumo en kWh en 15 minutos.
- **Instalaciones**: Energía eléctrica total consumida en el taller en cuestión. La variable mide el consumo en kWh en 15 minutos.
- **Aire comprimido**. La variable mide el consumo en kWh en 15 minutos.
- **Linea de producción 1**: Número de elementos producidos por la linea 1. En este caso sería el numero de coches de cierto modelo producido.
- **Linea de producción 2**: Número de elementos producidos por la linea 2. En este caso sería el numero de coches de cierto modelo producido.
- **Linea de producción 3**: Número de elementos producidos por la linea 3. En este caso sería el numero de coches de cierto modelo producido.
- **Temperatura interna**: Temperatura ambiente del taller. La variable mide la temperatura interna media en ºC en 15 minutos.
- **Temperatura externa**: Temperatura ambiente del exterior de la fábrica. La variable mide la temperatura externa en ºC en 15 minutos.

Todas estas medidas se toman con una frecuencia de 15 minutos y corresponden al espacio temporal que comprende desde Enero de 2016 hasta Diciembre de 2018. A continuación se presentan un par de tablas con el resumen estadístico del conjunto de datos. En la tabla 2.1 se representan los datos en bruto y en la tabla 2.2 los datos omitiendo el nivel de 0, debido a la bimodalidad de la distribución. Destacar que estos valores 0 no son *missing data* sino el valor para cierto consumo que en ese momento es 0, es decir se encuentra en un modo apagado.

	Pot.	Per.	Des.	G. dir.	G. tec.	G. cal.	Cli.	Frio	Alu.	Inst.	A. comp.	L.1	L.2	L.3	T. int.	T. ext.
Count	100608	100608	41431	100608	101472	100704	71839	101472	101472	101472	101472	89056	89056	89056	100992	101472
Mean	0.92	0.00	0.06	66.25	59.48	80.85	33.75	40.07	63.06	315.30	4.52	5.34	4.51	3.00	23.06	17.44
Std	1.38	0.05	0.25	70.06	81.00	94.14	24.18	76.66	30.47	160.73	5.19	6.30	6.01	4.04	2.25	7.43
Min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	12.70	-1.80
25 %	0.00	0.00	0.00	0.00	1.00	20.00	3.00	0.51	39.00	142.87	0.00	0.00	0.00	0.00	22.00	11.63
50 %	1.00	0.00	0.00	0.00	22.50	46.00	35.00	1.00	80.00	357.38	2.77	2.00	0.00	0.00	22.70	17.20
75 %	1.00	0.00	0.00	141.48	84.00	112.00	60.51	1.00	84.00	439.16	6.93	11.00	10.00	6.00	23.80	22.97
max	25.00	1.00	2.00	282.96	681.00	1466.00	126.05	276.00	155.00	884.49	13.86	60.00	59.00	58.00	32.20	39.50

Tabla 2.1: Se representa los estadísticos básicos de cada variable del *dataset*. Entre ellos encontramos el número de observaciones, la media, la desviación estándar de la media, el valor mínimo, el valor máximo y los percentiles de 25,50, y 75 %.

	Pot.	Per.	Des.	G. dir.	G. tec.	G. cal.	Cli.	Frio	Alu.	Inst.	A. comp.	L.1	L.2	L.3	T. int.	T. ext.
Count	100608	100608	41431	100608	101472	100704	71839	101472	101472	101472	101472	89056	89056	89056	100992	101472
Mean	1.80	1.0	1.00	134.61	78.26	95.67	35.09	52.93	68.84	315.44	7.92	9.96	10.16	6.21	23.06	17.44
Std	1.46	0.0	0.03	27.78	84.63	95.24	23.68	84.15	24.81	160.63	4.49	5.30	4.90	3.72	2.25	7.43
Min	1.00	1.0	1.00	11.79	1.00	1.00	0.01	0.25	2.00	0.07	1.39	1.00	1.00	1.00	12.70	-1.80
25 %	1.00	1.00	1.000	106.11	15.00	31.00	9.00	1.00	64.00	143.01	4.16	6.00	7.00	4.00	22.00	11.63
50 %	1.00	1.00	1.00	141.48	39.00	60.00	36.51	1.00	81.00	357.46	5.54	10.00	10.00	6.00	22.70	17.20
75 %	2.00	1.00	1.00	141.48	126.00	128.00	61.00	120.89	84.00	439.19	13.86	13.00	13.00	8.00	23.80	22.97
Max	25.00	1.00	2.00	282.96	681.00	1466.00	126.05	276.000	155.0	884.4	13.8	60.0	59.0	58.0	32.2	39.5

Tabla 2.2: Se representa los estadísticos básicos de cada variable del *dataset*. Entre ellos encontramos el número de observaciones, la media, la desviación estándar de la media, el valor mínimo, el valor máximo y los percentiles de 25,50, y 75 %. Todos los estadísticos son calculados omitiendo los valores 0 en cada serie.

2.2 | Preprocesado

2.2.1. *Missing data*

Observando las dos tablas 2.1 y 2.2 podemos ver que cada serie tiene un número de observaciones no homogéneo (valores de Count), esto se debe a que esta tabla es generada con una función que directamente omite los NaNs del cálculo. Es decir, tenemos *missing data* en nuestro conjunto de datos.

En ocasiones (la mayoría) se pueden encontrar conjuntos de datos que representan series temporales con momentos o rangos temporales donde no hay medida. Es de gran importancia un tratamiento acertado de estos *missing data* ya que hay modelos de regresión que son altamente sensibles a estos, aunque también hay modelos que permiten la presencia de *missing data* pero que en el caso de series temporales no es lo adecuado porque romperíamos las secuencias.

En 1976 Rubin [19, 20] presentó una clasificación para los *missing data* que se usa actualmente. Este sistema describe la relación entre los datos y la probabilidad de un valor faltante. Para comprender y describir las distribuciones de probabilidad de los distintos mecanismos debemos introducir la teoría básica y la notación mas usada en la literatura. Sea $Y = (y_1, \ldots, y_n)^T$ que denota un vector de una variable que corresponde con todos los datos, incluyéndose los validos y los faltantes. Además se tiene en cuenta el indicador de dato faltante $M = (M_1, \ldots, M_n)^T$ que define una variable binaria que representa si el valor es valido o es un dato faltante.

En esencia la teoría de Rubin comprende la colección de datos de una variable como una matriz de dos vectores: El primero que recoge el valor de la observación y el segundo corresponde con el indicador M. En la practica resulta imposible conocer la función de distribución de M pero la naturaleza de esta va a determinar los mecanismos que están definidos en base a la probabilidad de distribución condicional de M dado los datos completos $f(M|Y, \phi)$, siendo ϕ un vector desconocido que describe la probabilidad de los *missing data*.

- *Structurally missing data*. Existe una razón lógica por la que estos datos faltan. Por ejemplo, unas medidas se empiezan a tomar en momentos temporales distintos.

- *Missing completely at random.* No existe una razón lógica a la falta del dato. Si sabemos que no existe una relación con el resto de medidas (propias o de otras variables). Es decir, este dato no puede ser reproducido con los datos validos.

$$P\left(M|Y_{obs}, Y_{mis}, \phi\right) = P(M|\phi) \quad \text{for all} \quad Y, \phi \tag{2.1}$$

- *Missing at random.* A diferencia del anterior, los datos si pueden mantener una relación con los restantes. En este caso si se podría reproducir los datos con los válidos mediante algún modelo de imputación.

$$P\left(M|Y_{obs}, Y_{mis}, \phi\right) = P\left(M|Y_{obs}, \phi\right) \quad \text{for all} \quad Y_{mis}, \phi \tag{2.2}$$

- *Missing not at random.* Los *missing data* tienen relación con datos de la misma variable.

$$P(M|Y, \phi) = P\left(M|Y_{obs}, Y_{mis}, \phi\right) \tag{2.3}$$

Una vez explicados los conceptos básicos se aplicaron una serie de análisis básicos basados en varias visualizaciones sobre los datos para "desentrañar" el origen de los *missing data* en las series. Paralelamente a este procedimiento se barajan los procesados a realizar en cada posible caso.

2.2.1.1. No hacer nada

La decisión mas sencilla. Basta con seleccionar nuestros algoritmos de predicción adecuadamente ya que existen varios que son capaces de operar con *missing data*, o bien estimándolos implícitamente o ignorándolos. Debido a que nuestro objetivo relaciona modelos muy distintos y algunos no van a poder trabajar con los *missing data* descartamos esta opción.

2.2.1.2. Estimación usando la media o mediana

Este procedimiento consiste en calcular la media/mediana de los datos validos de la serie y reemplazar los *missing data* por este valor. Este procedimiento tiene varias desventajas que se hacen patentes cuando se quiere modelar una predicción con el *dataset*. En primer lugar no tiene en cuenta el resto de features y en un *dataset* como el que tenemos esto es básico. Además en *datasets* muy grandes la precisión de la imputación es malísima, solo basta con imaginarse como desvirtuariamos la distribución de una serie con un 50 % de *missing data*.

2.2.1.3. Interpolacion

Este procedimiento localiza los puntos donde se encuentran los *missing data* y "busca" a su alrededor para estimar por interpolación. De nuevo este procedimiento no tiene en cuenta el resto de features y si observamos nuestra

distribución temporal no parece el mas indicado ya que nosotros disponemos de unos *missing data* homogéneos en ciertos espacios temporales, no son puntos aislados.

2.2.1.4. *Listwise*

En general existen varios métodos mas pero en este punto resulta demasiado ambicioso para el global del estudio hacer un tratamiento mas extenso. De igual forma hemos visto que la mayoría de los métodos de estimación analizados no son adecuados para nuestro *dataset*, o al menos son muy arriesgados dadas las circunstancias de éste. Por ejemplo, no podemos interpolar un espacio temporal de 10000 observaciones y sería arriesgado depender de una imputación sobre esta variable para usarla como predictor de un modelo de regresión. Debido a ello se va a optar por eliminar las dos variables con mas de un 25 % de *missing data*: Permeada y Climatizadores. A su vez, nos apoyamos en su bajo peso en las correlaciones con las demás. Además, vamos a borrar la variable desionizada pues no tiene ningún peso aparente en el análisis multivariante y, además, no es una variable muy interesante para su predicción según el objetivo propuesto desde la fábrica. Aunque no es una decisión tomada por ser una variable con missing data, también se ha eliminado la variable Aire Comprimido debido a un cambio de escala transcurrido a mediados de julio de 2017. Este cambio de escala no se puede invertir fácilmente debido a como se toman las medidas. Por ejemplo: Al detectarse que el caudalímetro llega a 1 m^3 se genera un pulso a la entrada del PLC.

Nuestro consumo de agua potable de forma instantánea es C pero el consumo reflejado en nuestro aparato de medida será el número de veces que se supere cierto umbral de medida en el periodo de tiempo de muestreo (15 minutos).

Respecto al resto de variables no nos arriesgamos en exceso si consideramos que nos encontramos ante una *missing data* de cáracter estructural ya que siempre aparece en "bloques" y al principio o al final del *dataset*. Por esta disposición optamos por tomar la senda de *listwise* que consiste en borrar todos los registros de cada serie para un momento o rango temporal. Esto lo podemos realizar sin tomar ninguna precaución porque no se va a romper la estructura temporal de la serie ya que nos hemos precavido con nuestro análisis preliminar.

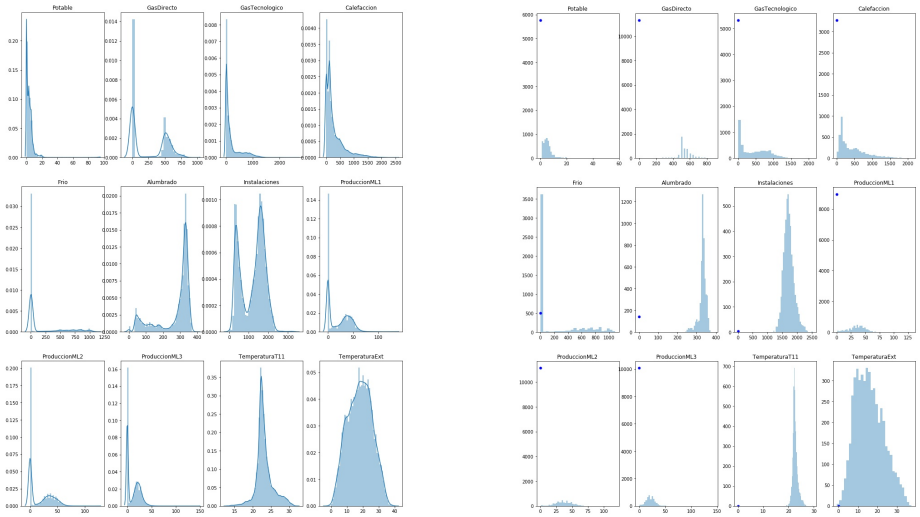
A continuación se presenta la tabla 2.3 con la estadística básica del *dataset* en la que ya se han eliminado la *missing data* y las variables no necesarias.

A diferencia de las tablas anteriores en esta ya podemos compprobar que el problema de *missing data* ha sido resuelto.

Con el fin de mostrar el carácter bimodal de la mayoria de variables del *dataset* se muestra la gráfica 2.1

	Pot.	G. dir.	G. tec.	G. cal.	Frio	Alu.	Inst.	L.1	L.2	L.3	T. int.	T. ext.
Count	88192	88192	88192	88192	88192	88192	88192	88192	88192	88192	88192	88192
0-Freq	44100	47371	23381	14961	24011	618	44	41082	49062	45667	0	0
Mean	1.76	137.79	72.09	84.16	63.19	68.30	301.87	9.95	10.15	6.22	23.17	18.15
Std	1.41	28.33	83.38	85.23	88.50	24.86	153.29	5.31	4.91	3.73	2.35	7.53
Min	1.00	11.79	1.00	1.00	0.25	5.00	0.07	1.00	1.00	1.00	12.70	-1.80
25 %	1.00	106.11	13.00	29.00	1.00	63.00	133.86	6.00	7.00	4.00	22.00	12.40
50 %	1.00	141.48	33.00	51.00	1.00	81.00	346.75	10.00	10.00	6.00	22.7	18.30
75 %	2.00	141.48	111.00	113.00	144.03	84.00	425.05	13.00	13.00	8.00	24.00	23.729
max	25.00	282.96	681.00	845.00	276.00	153.00	884.49	60.00	59.00	58.00	32.20	39.50

Tabla 2.3: Se representa los estadísticos básicos de cada variable del *dataset*. Entre ellos encontramos el número de observaciones, la media, la desviación estándar de la media, el valor mínimo, el valor máximo y los percentiles de 25,50, y 75 %. Todos los estadísticos son calculados omitiendo los valores 0 en cada serie.



(a) .. (b) ..

Figura 2.1: Se representa las distribuciones de los datos. En (a) se representan los datos en bruto. Por el contrario, en (b) se ha eliminado los valores de 0 que corresponderían con el modo apagado del sistema.

2.2.2. *Outliers*

Uno de los primeros pasos a la hora de un correcto análisis exploratorio del *dataset* es realizar una detección de *outliers*. Aunque los outliers suelen ser considerados como ruido o errores de medida en ocasiones representan información mucho mas importante. Por ejemplo pueden dar una información sobre un inminente fallo grave en un sensor de medida que esta empezando a registrar datos atípicos. No etiquetar estos *outliers* y tratarlos pueden llevar a escoger un modelo de forma errónea, una estimación de parámetros con *bias*. [21, 22]

En la literatura se encuentran varias definiciones sobre este conceptos. Estas diferencias en la definición se debe a los supuestos ocultos que se toman de

la estructura de los datos y los métodos de detección. Una de las definiciones mas completas es la dada por Hawkins en [23]: *An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism.*

En nuestro ejemplo no hemos sido muy ambiciosos a la hora de proponer diferentes formas de evaluar estos *outliers* (ver otros procedimientos en [24, 25]). En realidad se ha optado por la forma más sencilla que consiste en calcular el rango intercuartílico $\text{IQR} = Q_3 - Q1$ y usar la siguiente regla: Los candidatos a *outliers* son los puntos que están por encima de $Q_3 + 1{,}5 \cdot \text{IQR}$ o por debajo de $Q_1 - 1{,}5 \cdot \text{IQR}$ sugerida por Tukey en [26]. Esto queda reflejado en la figura 2.2.

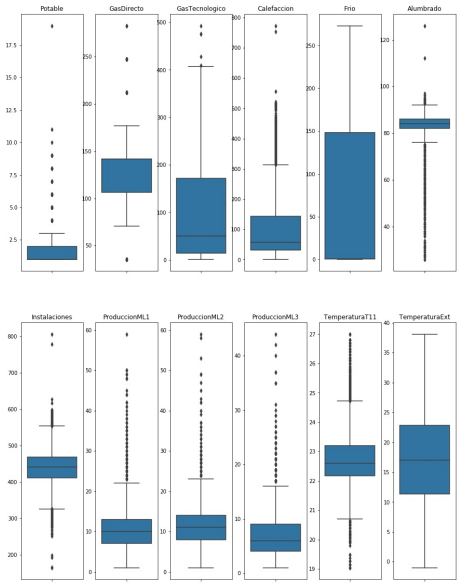


Figura 2.2: Gráfica de cajas y bigotes representando la cantidad de *outliers* mediante la regla anterior.

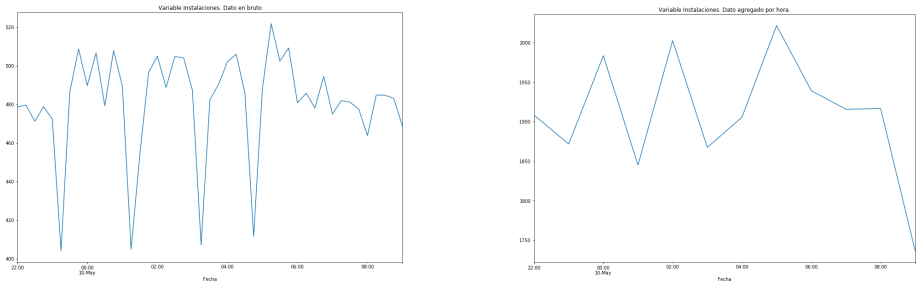
Una vez se ha logrado una caracterización de estos candidatos se toma la decisión de comunicarlo en la fábrica. La respuesta obtenida es que estos puntos son datos reales así que no hemos realizado ningún tratamiento especial sobre estos puntos.

Las tareas de preprocesado de los datos han concluido aquí. Este preprocesado es muy preliminar, de tal forma que nos permita realizar pruebas de distintos modelos candidatos y así poder evaluar, en un futuro, un preprocesado en mayor profundidad.

A pesar de que el preprocesado ha concluido dedicamos el apartado siguiente a hablar de la agregación de los datos realizadas que, aunque no se trata de una labor de preprocesado general, podemos englobar dentro de este capítulo en base a que responde al problema concreto de predicción que tenemos aquí.

2.2.3. Agregación

Debido al error de muestreo presente en los datos provocado por las medidas analógicas generadas por los sistemas de medida encargados en ello. Por ello se toma la decisión de agregar los datos por hora, tomándose la suma para las variables de consumo y producción y el valor mediano para las variables de temperatura. Además de subsanar nuestros problemas de muestreo de los datos nos va a aportar una menor cantidad de datos que permitirá una mayor agilidad computacional, tanto a la hora de entrenar el modelo como de un potencial despliegue en funcionamiento *online*. Se representa en la figura 2.3 una pequeña muestra del cambio al tomar la agregación de los datos a un hora.



(a) (b)

Figura 2.3: La figura (a) representa una muestra de la variable instalaciones con sus datos en bruto, es decir con un paso temporal de 15 minutos. Por otro lado en (b) se representa la misma muestra pero con los datos agregados por hora.

Finalmente se muestra los estadísticos básicos sobre el conjunto de datos final en la tabla 2.4.

	Pot.	G. dir.	G. tec.	G. cal.	Frio	Alu.	Inst.	L.1	L.2	L.3	T. int.	T. ext.
Count	22048	22048	22048	22048	22048	22048	22048	22048	22048	22048	22048	22048
0-Freq	5769	11309	5320	3271	501	142	3	8981	11137	10077	0	0
Mean	4.76	523.75	279.31	328.21	188.21	273.05	1207.05	35.85	36.41	22.10	23.16	18.14
Std	4.40	126.29	330.16	336.74	322.49	97.45	606.90	15.08	13.57	8.83	2.35	7.53
Min	1.00	35.37	1.00	1.00	1.00	33.00	7.36	1.00	1.00	1.00	12.70	-1.55
25 %	2.00	495.18	51.00	112.00	2.00	260.00	537.30	27.00	28.00	17.00	22.01	12.40
50 %	4.00	530.55	125.00	193.00	3.00	321.00	1413.20	37.00	36.00	22.00	22.73	18.30
75 %	6.00	565.92	423.00	444.00	341.50	335.00	1684.54	46.00	46.00	27.00	24.02	23.74
Max	94.00	919.62	2663.00	2515.00	1088.00	391.00	3314.24	131.00	120.00	144.00	32.15	39.20

Tabla 2.4: Se representa los estadísticos básicos de cada variable del *dataset*. Entre ellos encontramos el número de observaciones, la media, la desviación estándar de la media, el valor mínimo, el valor máximo y los percentiles de 25,50, y 75 %. Todos los estadísticos son calculados omitiendo los valores 0 en cada serie.

2.3 | Series temporales

En general, los modelos de series temporales pueden ser univariantes o multivariantes. En el primer caso, se analiza una única serie temporal basándonos en su histórico. En cambio, en el caso de un modelo multivariante estudiaríamos varias series temporales a la vez debido a que se considera estrechas dependencias entre distintas variables. A continuación, se va a detallar los distintos aspectos matemáticos de este concepto.

2.3.1. Definición

Una serie temporal es una secuencia de N, observaciones ordenadas cronológicamente, sobre una característica (serie univariante) o sobre varias características (serie multivariante).

2.3.1.1. Representación matemática de series temporales univariantes

$\boldsymbol{X}(t) = \{x_t : t \in T\}$, donde x_t es la observación t, $(1 \leq t \leq N)$ de la serie y N es el número de observaciones. Estas observaciones se pueden recoger en un vector $\boldsymbol{X}(t) \equiv [x_1, x_2, ..., x_N]$

2.3.1.2. Representación matemática de series temporales multivariantes

$\boldsymbol{X}(t, x_1, x_2, ..., x_M) = \{\boldsymbol{x}_t : t \in T\}$, donde $\boldsymbol{x}_t \equiv [x_{t1}, x_{t2}, ..., x_{tM}]$ es la observación $t, (1 \leq t \leq N)$ de la serie y N es el numero de observaciones. Estas observaciones se pueden recoger en una matriz de orden N x M.

$$\mathbf{X} \equiv \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix} \equiv \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1M} \\ x_{21} & x_{22} & \dots & x_{2M} \\ \vdots & \vdots & & \vdots \\ x_{N1} & x_{N2} & \dots & x_{NM} \end{bmatrix} \tag{2.4}$$

2.3.2. Componentes

Un estudio descriptivo consiste en descomponer la serie temporal en una serie de componentes primarias. Esto, en ocasiones, no resulta demasiado efectivo pero se un buen punto de partida a la hora de comenzar la comprensión de la serie. Podemos descomponer la serie en cuatro componentes principales.

- **Tendencia**: Se puede definir como el cambio temporal de la media.

- **Componente cíclica**: Esta componente refleja comportamientos recurrentes, aunque no tienen por qué ser exactamente periódicos. Estos ciclos se refieren a términos de medio y largo alcance así que los ciclos, aunque no están determinados por un periodo único, son de duraciones superior al año.
- **Componente estacional**: Esta componente consiste en las fluctuaciones existentes en cierto subgrupo de un periodo mayor. Por ejemplo, numero de turistas en Verano en Cantabria.
- **Componente Irregular**: Esta componente se debe a todas aquellas fluctuaciones a causa de variables desconocidas y estocásticas.

Dependiendo de como combinemos las componentes de la serie podemos distinguir tres modelos adecuados para representar la serie temporal.

- Modelo aditivo.
$$X(t) = T(t) + S(t) + C(t) + I(t) \tag{2.5}$$
- Modelo multiplicativo.
$$X(t) = T(t) \cdot S(t) \cdot C(t) \cdot I(t) \tag{2.6}$$
- Modelo mixto.
$$X(t) = T(t) \cdot S(t) \cdot C(t) + I(t) \tag{2.7}$$

Donde $X(t)$ es la observación y $T(t)$, $S(t)$, $C(t)$, $I(t)$ son respectivamente las componentes de tendencia, estacional, cíclica e irregular. El modelo multiplicativo no asume la independencia de cada término y, por ello, permite la influencia de cada uno en los demás. Por otro lado el mixto suele mantener la independencia del término estocástico irregular como independiente de las demás componentes. En general, escogeremos una descomposición aditiva cuando la variación estacional sea constante. Por otro lado, si encontramos que la variación estacional varia con el tiempo el modelo multiplicativo reflejaría mejor la realidad.

2.3.3. Proceso estocástico

Como hemos descrito anteriormente, la componente Irregular corresponde a un proceso estocástico y esto es lo que permite que las series temporales resulten muy complicadas de modelizar. Un proceso estocástico es una secuencia de variables aleatorias ordenadas cronológicamente referidas a una o varias características de una unidad observable en diferentes momentos.

2.3.3.1. Representación matemática de procesos estocásticos univariantes

Y_t, donde Y_t es una variable aleatoria escalar referida a la unidad observable considerada en el momento t.

2.3.3.2. Representación matemática de procesos estocásticos multivariantes

$\boldsymbol{Y}_t$, donde $\boldsymbol{Y}_t \equiv [y_{t1}, y_{t2}, ..., y_{tM}]$ es una variable aleatoria vectorial referida a la unidad observable considerada en el momento t.

2.3.4. Clasificación

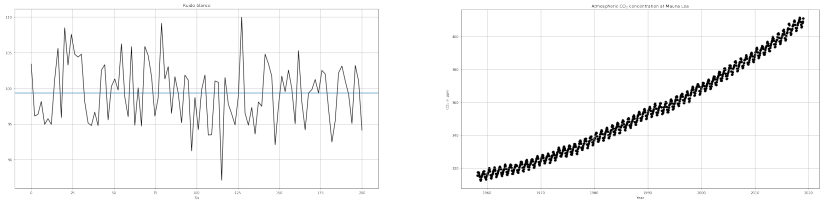
Es importante clasificar las series temporales en base a como se comportamiento temporal de las propiedades estadísticas.

2.3.4.1. Procesos estacionarios

Un proceso estocástico es estacionario cuando las propiedades estadísticas de cualquier subsecuencia finita $x_{t_1}, x_{t_2}, ..., x_{t_n}$, $n >= 1$ de componentes de (X_t) son semejantes a las de cualquier otra secuencia $x_{t_1+h}, x_{t_2+h}, ..., x_{t_n+h}$, para cualquier numero entero $|h| = 1, 2, \ldots$

2.3.4.2. Procesos no estacionarios

Un proceso estocástico es no estacionario cuando las propiedades estadísticas de al menos una subsecuencia finita $x_{t_1}, x_{t_2}, ..., x_{t_n}$, $n >= 1$ de componentes de (X_t) son diferentes a la de la secuencia $x_{t_1+h}, x_{t_2+h}, ..., x_{t_n+h}$, para al menos algún numero entero $|h| = 1, 2, \ldots$. Podemos ver un ejemplo de esto en la figura 2.4.



(a) (b)

Figura 2.4: Series estacionaria y no estacionaria. En (a) se representa una serie estacionaria que se trata de un ruido blanco simulado con una media $\mu = 100$ y $\sigma = 5$. En (b) se representa una serie no estacionaria que se trata de la concentración de CO_2 en la atmósfera. Fuente: [27].

2.4 | Análisis de series temporales

2.4.1. Visualización de datos

En primer lugar se puede recurrir a la representación gráfica de los datos para obtener un cierto conocimiento de éstos. La visualización de datos es la presentación gráfica de información con dos propósitos. Por un lado, la interpretación y construcción de significado a partir de los datos (es decir, el análisis); y por otro lado, la comunicación. La visualización de datos es una herramienta indispensable para descubrir y comprender la lógica que opera detras de un conjunto de datos, así como para comunicar esta información de forma sencilla y eficaz, sobre todo en determinados conceptos que resulta más sencilla la comunicación visual que la verbal.

Históricamente la visualización se ha desarrollado de forma intrínseca a los datos. Sin embargo, a finales del siglo XVIII y principios del XIX es cuando aparecen los primeros estudios sobre visualización de datos y su importancia en la reconstrucción de los fenómenos subyacentes. En este sentido, hay que destacar el trabajo pionero del economista escocés William Playfair [28]. Éste es considerado el fundador de aplicación de técnicas gráficas para el análisis estadístico, inventando gráficos como los de líneas, áreas (como en la figura 2.5), barras y de tarta.

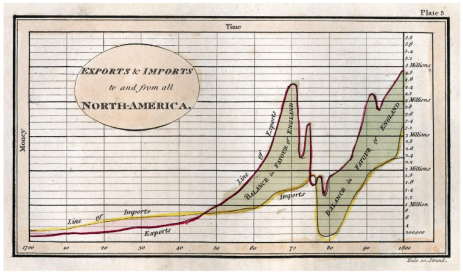
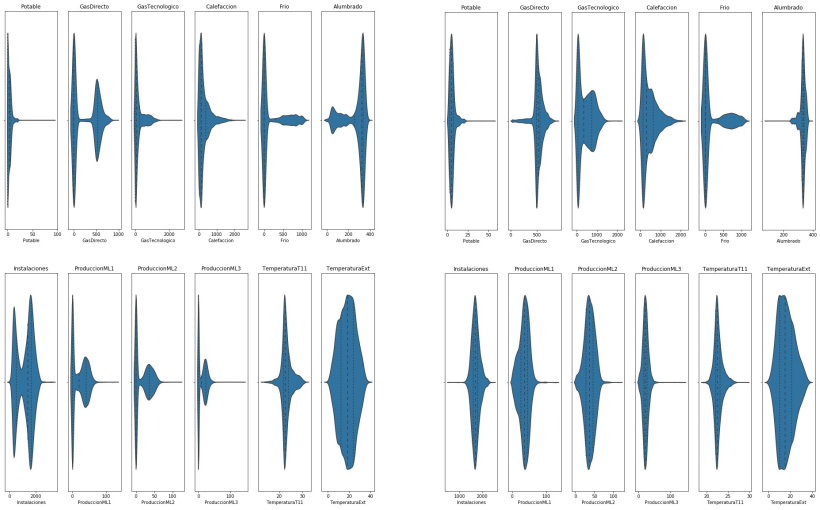


Figura 2.5: Balance importaciones-exportaciones norteamericanas. Se observa un gráfico de líneas y uno de áreas subyacente a éste reflejando el balance respecto a Inglaterra. Fuente: [28].

A continuación vamos a mostrar una serie de visualizaciones sobre nuestros datos. En primer lugar y de forma condensada se representan las distribuciones de los datos con un gráfico de violines en la figura 2.6.

El diagrama de violín es la combinación del diagrama de cajas y bigotes clásico (visto en la figura 2.2 y un diagrama de densidad de probabilidad simétrico. En el interior del diagrama se representan los distintos cuartiles. Como se puede ver en este punto los diagramas de cajas y bigotes están muy limitados para la visualización de los datos, ya que debido a su simplicidad no nos aporta ninguna idea de como se distribuye el conjunto de datos. Más concretamente, si observamos la figura 2.2 no podríamos ver el efecto de la bimodalidad que introducen los valores cero del estado apagado en nuestros datos.

A continuación se representa otra clásica visualización sobre posibles estacio-



(a) (b)

Figura 2.6: Diagramas de violines para cada variable. (b) corresponde con el diagrama violín omitiendo los valores de cero. Se hace ver como se corrige un poco la bimodalidad introduce la dualidad "encendido/apagado" en la serie.

nalidades dentro de la misma semana para la variable de unidades producidas en la línea 1 donde se puede ver un comportamiento distinto para los días de fin de semana (figura 2.7).

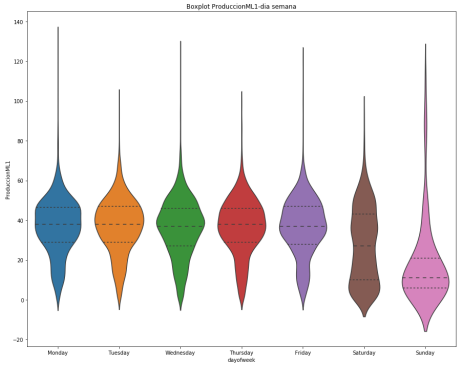


Figura 2.7: Se representa la gráfica de violines de la linea de producción 1 en agregaciones diarias en las que se muestra un componente estacional.

2.4.2. Descomposición de la serie

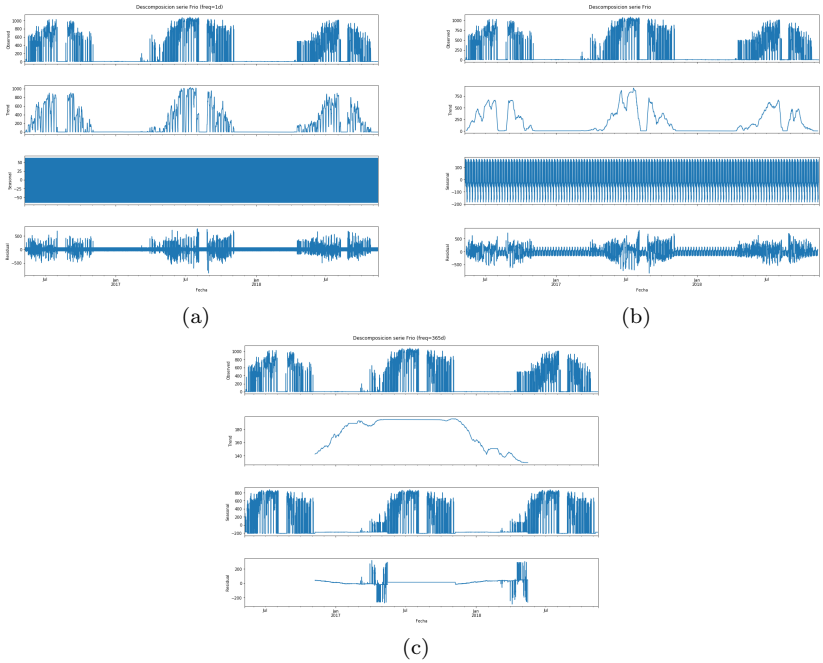
Siguiendo el enfoque comentado en el apartado referente a la serie temporal y sus componentes se ha procedido a la descomposición de las series en sus términos. Al igual que en todos y cada uno de los procesos descritos anteriormente podríamos realizar una disertación completa sobre este asunto y abordar las

distintas formas de proceder para este objetivo [29, 30].

El método de descomposición clásico se originó en la década de 1920. Es un procedimiento relativamente simple, y constituye el punto de partida para la mayoría de los otros métodos de descomposición de series de tiempo [29, 30]. A continuación se describirá de forma cualitativa en los pasos a seguir.

1. Identificar el tipo de modelado: Se recuerda que escogeremos un modelo multiplicativo cuando la variación estacional varía con el tiempo.
2. Identificar la tendencia: Se calcula a través de un filtro de medias móviles.
3. Identificar la componente estacional: De forma sencilla se haya restando la componente de tendencia y agrupando los datos en una frecuencia estacional y promediándolos. Esta frecuencia puede partir de una intuición como puede ser la estacionalidad del turismo de sol.
4. Identificar la componente irregular: Es la señal que resulta de restar del original la componente de tendencia y de estacionalidad.

Como ejemplo de esto tenemos la figura 2.8 donde se muestra la descomposición en base a la frecuencia estacional que esperamos ver.



(a) (b)

(c)

Figura 2.8: Se representa la descomposición de la serie Frío. En (a) se ha tomado como frecuencia estacional un día. En (b) tomamos como frecuencia una semana. Y, finalmente, en (c) se toma como frecuencia un año.

2.4.3. Estacionariedad

Ya se ha descrito anteriormente en que consiste el concepto de estacionariedad pero no hemos explicado como podemos determinarlo de forma efectiva y por qué es importante.

Como se ha descrito la estacionariedad de un proceso consiste en que sus propiedades estadísticas se mantienen constantes en el tiempo. Esto no significa que la serie no cambie, solo que la forma en la que cambia no varía con el tiempo.

En general, una serie estacionaria es mucho más fácil de predecir, incluyendo la idea de que muchos de los métodos sencillos parten de esta suposición. Si se comportaba de una manera en el pasado, podríamos suponer que se seguirá comportando de la misma forma en el futuro (más correctamente: tiene una gran probabilidad de continuar comportándose de la misma forma).

En primer lugar para detectar la estacionariedad podemos usar la descomposición de la serie en sus componentes y de forma visual obtener nuestra respuesta. De forma mas metódica tenemos a nuestra disposición algunos test de hipótesis que se encargan de probar la estacionariedad de la serie. El más usado es el test de Dickey-Fuller que busca determinar la existencia o no de raices unitarias en la serie temporal. La hipótesis nula H_0 de esta prueba es que existe una raiz unitaria. El planteamiento mas sencillo del test Dickey-Fuller es el siguiente:

$$y_t = \mu + \rho y_{t-1} + \varepsilon_t \tag{2.8}$$

donde μ y ρ son parámetros a estimar y ε_t es un termino de error que se asume cumple las propiedades de ruido blanco. Dado que ρ es un coeficiente de autocorrelación toma valores $-1 < \rho < 1$. Si $\rho = 1$, la serie y es no estacionaria. De esta forma, la hipótesis de estacionariedad se puede evaluar analizando si $\rho < 1$. Así, el test plantea contrastar estadísticamente si $\rho = 1$.

Los resultados sobre la variable de la energía consumida en las instalaciones arrojan un $p - value = 1 \cdot 10^{-23}$. Esto parece bastante absurdo y se ha propuesto poner a prueba al test con una serie simulada no estacionaria del mismo tamaño. En la figura 2.9 se representa la serie simulada.

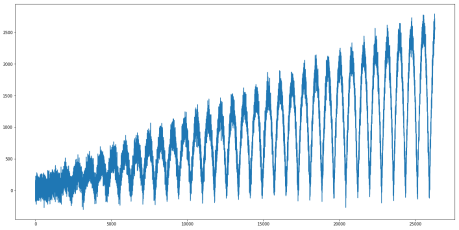


Figura 2.9: Representación serie simulada. Se puede comprobar la evidente falta de estacionariedad.

Y en la tabla 2.5 se representan los resultados del test en ambas muestras.

Podemos ver que incluso con una serie claramente no estacionaria el test arroja un resultado que confirmaría la estacionariedad de la serie. Esto se

	Test Statistic.	p-value.	Lags.	Critical value 1 %	Critical value 5 %	Critical value 10 %
Instalaciones	-12.588	0.000	47	-3.500	-2.862	-2.567
Simulada	-7.295	$1{,}388 \cdot 10^{-}35$	49	-3.431	-2.862	-2.567

Tabla 2.5: Principales resultados de los estadísticos correspondientes al test de Dickey-Fuller sobre la serie Instalaciones real y la serie simulada.

debe a que ambas series son de un tamaño elevado ($>$ 10000 datos) y surgen problemas con los test estadísticos. En muestras muy grandes, los valores de p bajan rápidamente a cero y no aportan ningún resultado sobre la significancia estadística [31].

En este sentido vamos a trabajar sobre la estacionariedad de la serie en el capitulo posterior ya que estará íntimamente relacionado con los modelos predictivos clásicos.

3 | Modelos clásicos para predicción de series temporales

Enlazando lo expuesto en el apartado anterior debemos tratar la serie que tenemos porque a pesar del test estadístico podemos observar que no es estacionaria. En este sentido se nos abren numerosas alternativas para lograr esta estacionariedad.

3.1 | Metodología Box-Jenkins

En el análisis de series temporales, la metodología Box-Jenkins, nombrada en honor a los estadísticos Geroge Box y Gwilym Jenkins, se aplica a los modelos autorregresivos de media móvil ARMA o a los modelos autorregresivos integrados de media movil ARIMA o su variante estacional SARIMA para encontrar el mejor modelo posible basado en los datos disponibles.

El método original utiliza un enfoque de modelado iterativo en tres etapas, usando datos de un horno de gas. Estos datos son conocidos como datos de Box-Jenkins del horno de gas para la evaluación comparativa de modelos de predicción.

Las tres etapas del modelado iterativo son las siguientes:

- Identificación y selección del modelo: Se debe asegurar que las series son estacionarias. Se puede partir de la identificación de la estacionalidad de la serie dependiente y el uso de los gráficos de las funciones de autocorrelación y de autocorrelación parcial de la serie de tiempo se utilizan para decidir cuál componente se debe utilizar en el modelo, el promedio autorregresivo (AR) o un promedio móvil (MA).
- Estimación de parámetros usando algoritmos de cálculo para tener coeficientes que mejor ajusten el modelo ARIMA seleccionado. Los métodos más comunes usan estimación de máxima verosimilitud o mínimos cuadrados no lineales.
- Comprobar el modelo mediante el ensayo, si el modelo estimado se ajusta a las especificaciones de un proceso univariado estacionario. En particular, los residuos deben ser independientes el uno del otro, además, la media y la varianza deben ser constantes en el tiempo.

Como se comprueba la metodología Box-Jenkins es coherente con un desarrollo de una metodología dentro de cualquier problema de aprendizaje automático o aprendizaje estadístico.

3.1.1. Gráficas de autocorrelación

En primer lugar se presentan esta herramienta tan utilizada en el procesamiento de series temporales. La función de autocorrelación se define como la correlación cruzada de la señal consigo misma. En la mayoría de las series temporales estacionarias y sin los efectos de estacionalidad son útiles para estimar los parámetros básicos de los modelos autorregresivos y de media móvil [32].

La función de autocorrelación parcial es la misma función de autocorrelación en la que se elimina las dependencias intermedias, es decir si medimos la autocorrelación parcial para un lag k es la autocorrelación entre z_t y z_{t+k} con la dependencia lineal de z_{t+1} y z_{t+k-1} eliminada.

Para lograr unas gráficas de autocorrelación útiles se requiere convertir la serie en estacionaria y analizar las estacionalidades existentes. Este punto se representa problemático en estas series de grandes volúmenes de datos como se ha visto anteriormente con los test estadísticos de Dickey-Fuller. En este sentido debemos rechazar el uso de estos tests y optar, al menos por el momento, de realizar unas tareas básicas respecto a la tendencia y estacionalidad.

El primer paso es eliminar las posibles tendencias de la serie. Box-Jenkins recomiendan tomar la primera diferenciación de la serie [33].

$$y_t = x_{t+1} - x_t = \nabla x_{t+1} \tag{3.1}$$

Aunque en series muy problemáticas se puede recurrir a las diferenciaciones de orden 2 y siguientes [33].

$$\nabla^2 x_{t+2} = \nabla x_{t+2} - \nabla x_{t+1} = x_{t+2} - 2x_{t+1} + x_t \tag{3.2}$$

A su vez, debemos ser capaces de modelos los términos de estacionalidad presentes en la serie. En este sentido al tratarse de una serie de valores horarios podemos encontrar tres periodos estacionales destacados: un periodo diario, semanal, semestral (Invierno-verano) y anual. Para solventar estas estacionalidades se ha intentado realizar la diferenciación estacional para cada una.

Para este objetivo se trata de realizar una diferenciación estacional que es realizar el promedio de la serie en cada periodo sospechoso de albergar estacionalidad y substraerlo a la serie. Esto es lo que se ha realizado en primer lugar obteniendo una serie resultante en la que quedaba una estacionalidad persistente que no fuimos capaces de modelar. Esto es algo que puede suceder y no nos deja más remedio que optar por un modelo que permita modelar un carácter estacional en la serie, este modelo sería el SARIMA que explicaremos de forma resumida a continuación.

3.1.2. Modelo SARIMA

Como se ha comentado, en la práctica muchas series temporales contienen una componente estacional que resulta muy difícil de capturar. Este es el princi-

pal motivo para el cual se generalizo los modelos clásicos ARIMA integrando explícitamente un término de estacionalidad. La definición del modelo SARIMA resulta:

$$\phi_p(B)\Phi_P(B^s)W_t = \theta_q(B)\Theta_Q(B^s)Z_t \tag{3.3}$$

donde B denota el operador de lag, ϕ_p, Φ_P, θ_q, Θ_Q son polinomios de orden p,P,q,Q respectivamente. Z_t se refiere a un proceso puramente estocástico y el término W_t son diferenciaciones de la serie como se indica a continuación.

$$W_t = \nabla^d \nabla_s^D X_t \tag{3.4}$$

El modelo definido en las ecuaciones anteriores es lo que se denomina modelo SARIMA de orden (p,d,q)x(P,D,Q)s.

Debido a los problema sucedidos con la serie a la hora de eliminar la estacionalidad nuestras gráficas de autocorrelación y autocorrelación parcial (Figuras 3.1 y **??**) no nos indican ninguna estimación viable (no modelos con valores elevados de los parámetros p y q) y se ha optado por seguir el enfoque de Brockwell y Davis en la que optimizan la búsqueda del modelo mediante un *grid* de modelos y se prueban y validan con los conjuntos de entrenamiento y validación. Mediante esta metodología se obtiene un modelo SARIMA con parámetros (3,1,2)x(0,1,1)8

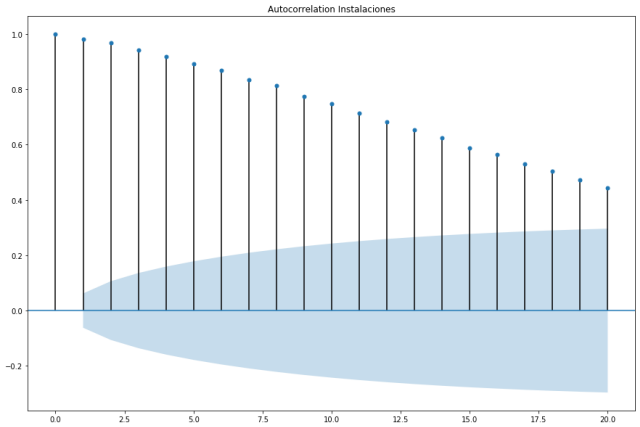


Figura 3.1: Gráfica de autocorrelación de la variable Instalaciones. Se puede ver un comportamiento exponencial decreciente pero que tarda en caer a 0, esto correspondería con una serie no estacionaria en la que posiblemente haya términos de estacionalidad no capturados.

Se reservan los resultados obtenidos para el modelo SARIMA para presentarlos en conjunto con el resto de métodos en los capítulos posteriores.

4 | Modelos de *machine learning* para predicción de series temporales

El *machine learning* es un subcampo de las ciencias de computación y matemática aplicada y, a su vez, una rama de la inteligencia artificial, cuyo objetivo es desarrollar técnicas que permitan que las computadoras aprendan. Se dice que un agente computacional aprende cuando su desempeño mejora con la experiencia [34]. De forma más concreta, se trata de encontrar algoritmos y heurísticas para convertir muestras de datos en programas de computadora, sin tener que escribir los últimos explícitamente. Los modelos o programas resultantes deben ser capaces de generalizar comportamientos e inferencias para un conjunto más amplio (potencialmente infinito) de datos.

Por lo tanto, nos encontramos con procesos de inducción de conocimiento. En muchas ocasiones el campo de actuación del aprendizaje automático se solapa con el de la estadística inferencial, ya que las dos disciplinas se basan en el análisis de datos. Sin embargo, el aprendizaje automático incorpora las preocupaciones de la complejidad computacional de los problemas.

4.1 | Tipos de aprendizajes

De forma resumida nos encontramos con 3 tipos de aprendizaje: supervisado, no supervisado y por refuerzo.

El aprendizaje supervisado es una técnica para deducir una función a partir de datos de entrenamiento. Los datos de entrenamiento consisten de pares de objetos (normalmente vectores): una componente del par son los datos de entrada y el otro, los resultados deseados. La salida de la función puede ser un valor numérico (como en los problemas de regresión) o una etiqueta de clase (como en los de clasificación). El objetivo del aprendizaje supervisado es el de crear una función capaz de predecir el valor correspondiente a cualquier objeto de entrada válida después de haber visto una serie de ejemplos, los datos de entrenamiento. Para ello, tiene que generalizar a partir de los datos presentados a las situaciones no vistas previamente.

El aprendizaje no supervisado tiene lugar cuando no se dispone de *outputs* para el entrenamiento. Sólo conocemos los datos de entrada, pero no existen datos de salida que correspondan a un determinado input. Por tanto, sólo podemos describir la estructura de los datos, para intentar encontrar algún tipo de organización que simplifique el análisis. Por ello, tienen un carácter exploratorio.

El aprendizaje por refuerzo es un área del aprendizaje automático, cuya

ocupación es determinar qué acciones debe escoger un agente computacional en un entorno dado con el fin de maximizar alguna noción de "recompensa" o premio acumulado.

4.2 | Algoritmos empleados

En este apartado se ha escogido una pequeña muestra de algoritmos sencillos y que pueden ser aplicados para este problema. Se tratará de dar un breve resumen de cada uno de los elegidos en base a su funcionamiento y la posibilidad de aplicación en la problemática que nos atañe.

4.2.1. Persistencia

En realidad, este modelo no debería estar presente en un apartado propio del *machine learning* pero por comodidad y homogeneidad se elige este punto como su introducción. Es un modelo *naive* ya que consiste en pronosticar el valor de la variable $Y(t+1) = Y(t)$.

4.2.2. Regresión lineal

La regresión lineal es un método usado para aproximar la relación existente entre una variable dependiente Y y una serie de variables dependientes X (siguiendo relaciones lineales).

$$Y_t = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon \tag{4.1}$$

donde $\beta_0, \beta_1, \beta_2, \cdots, \beta_p$ son los parámetros a ajustar, siendo β_0 el término constante o también denominado *intercept*. El término ε representa un elemento estocástico relacionado con el error asociado a las variables dependientes.

En concreto, el problema de regresión lineal busca el hiperplano óptimo que minimice cierta función de fitness, usualmente se trata de un problema de mínimos cuadrados ordinarios, es decir nuestra función de error se trata de un error cuadrático medio que aprovechamos para introducir como métrica de validación.

$$\text{mse} = \frac{1}{n} \sum_{i=1}^{n} \left(\hat{Y}_i - Y_i \right)^2 \tag{4.2}$$

siendo $\hat{Y}$ la estimación o predicción de la variable Y.

4.2.3. Máquinas de vectores soporte

Otro modelo de los empleados son las máquinas de vectores soporte cuyo contenido ha sido ampliamente relatado en el transcurso del master al que hace mención este trabajo. En este sentido, vamos a desarrollar la idea general del modelo de forma cualitativa.

Esta idea reside en transformar el espacio de entrada donde viven los datos de entrenamiento a un espacio de mayor dimensión mediante un mapeo no lineal donde podemos realizar una regresión lineal (similar a la vista en el punto anterior).

Es decir, crearemos virtualmente un problema de regresión lineal con un dimensión mucho mayor a la original que se resolverá con una regresión lineal que representará una relación de carácter no lineal en el espacio de entrada.

Para nuestro ejemplo hemos usado una versión optimizada en python que usa kernels lineales y su coste computacional es menor. [35]

En cuánto a la función de error se suele pasar de la norma L2 a una función denominada $\epsilon -$ insensitive definida de la siguiente forma $\text{máx}(0, |y - f(\mathbf{x})| - \epsilon)$ en la que permitimos un margen de libertad al modelo donde no penalizar los errores.

4.2.4. Perceptrón multicapa

Una red neuronal puede entenderse como una maquina bioinspirada en el funcionamiento del sistema nervioso para realizar una tarea. Dentro de este esquema la red neuronal esta formada por un conjunto de neuronas.

Para no extendernos demasiado con las definiciones y funcionamiento de las redes se tratara de resumir de forma rápida como se ha hecho con los modelos anteriores: Cada neurona recibe como entrada un conjunto de señales discretas o continuas, las pondera y transmite el resultado a las neuronas conectadas. Estos pesos guardan la mayor parte de la información sobre la red y el proceso por el cual se ajustan u optimizan estos pesos es el entrenamiento o aprendizaje.

El perceptrón multicapa es una arquitectura de red neuronal básica en la que juntamos varias capas de neuronas. Esta es la primera arquitectura de red neuronal propuesta y en 1986 se demuestra que un perceptrón mutlticapa con funciones de activación no lineales se trata de un aproximador universal [36].

Lo interesante de las arquitecturas basadas en redes neuronales es que son muy flexibles y diversas en cuanto a la forma de entrenar. Por ejemplo, nosotros podemos elegir las funciones de activación con la única condición de su derivabilidad para cumplir los algoritmos de optimización funcionales. Así como escoger una función de pérdida a nuestro gusto, en este sentido seleccionaremos como función de pérdida el error cuadrático medio *mse* ya que, como en los ejemplos anteriores, nos penaliza mas los errores mas grandes (es decir, hará que nuestra solución tienda a intentar ajustar mas los valores extremos, esta decisión es tomada siguiendo directrices de fabrica en la que no se consideran *outliers*

ningún valor de los datos).

4.2.5. Redes neuronales recurrentes

Las redes neuronales recurrentes son redes neuronales que presentan uno o más ciclos en el grafo definido por las interconexiones de sus unidades de procesamiento. La existencia de estos ciclos les permite trabajar de forma innata con secuencias temporales. Las redes recurrentes son sistemas dinámicos no lineales capaces de descubrir regularidades temporales en las secuencias procesadas y pueden aplicarse, por lo tanto, a multitud de tareas de procesamiento de este tipo de secuencias, en nuestro caso predicción de series temporales.

En la figura 4.1 podemos ver la estructura básica de una red neuronal recurrente con la que solventamos el problema de no conseguir transmitir cierta persistencia a lo largo del tiempo. Esto es la dependencia inherente de datos secuenciales respecto a su ordenamiento temporal.

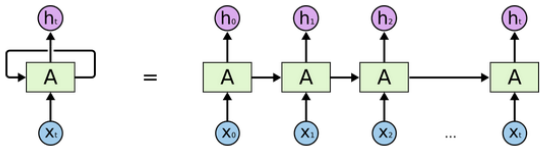


Figura 4.1: Representación simbólica de una red neuronal recurrente y su versión desenrollada. Fuente: [37]

Se pueden considerar estas redes como copias múltiples de la misma red, cada una de las cuales transmiten información a la siguiente. Este forma de entender las redes recurrentes como una cadena de neuronas nos aporta la intuición de que están íntimamente relacionadas con secuencias o listas. En efecto, son la arquitectura procedente del *deep learning* encargadas para modelar estos datos.

De hecho hay numerosos casos de éxito en el empleo de estas redes neuronales recurrentes para multitud de procesos secuenciales como el reconocimiento de voz, procesamiento de lenguaje natural [38], etc. Para la consecución de todos estos buenos desempeños juega un especial interés unas versiones especiales de redes recurrentes: las redes GRU y LSTM.

4.2.5.1. LSTM

En ocasiones solo necesitamos una información persistente a corto plazo para lograr la tarea actual de forma correcta. Por ejemplo. si tratamos de predecir la palabra siguiente a la secuencia "*Las estrellas están en el ...*" no necesitamos un contexto a largo plazo para determinar que con altísima probabilidad la siguiente palabra sería "*cielo*".

El problema reside cuando necesitamos tener a nuestra disposición un contexto más amplio (o varios contextos). Las redes neuronales recurrentes no parecen ser

capaces de aprender las dependencias de contextos a largo plazo [39].

Las redes LSTM, *long short-term memory*, son un tipo especial de red neuronal recurrente capaz de aprender y gestionar dependencias a corto y largo plazo [40]. Para entender este tipo de arquitectura debemos explicar muy resumidamente como es el funcionamiento interior de una de estas celdas LSTM.

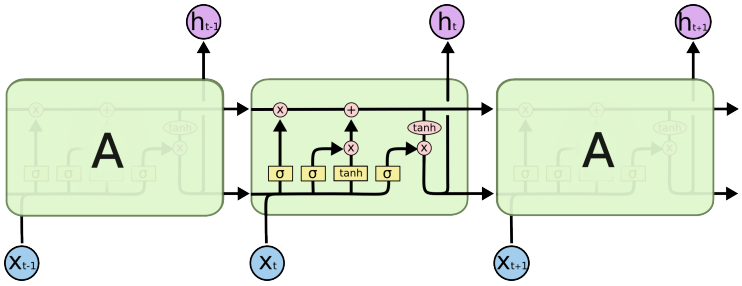


Figura 4.2: Representación de la estructura interna de una celda LSTM. Fuente: [37]

En la figura 4.2 se puede identificar 4 capas con las cuales tenemos la capacidad de eliminar o agregar información al estado de la celda. Las puertas son capas sigmoidales con una operación de multiplicación que tienen como función dejar pasar la información (Si el valor de salida es 1) o no (en caso contrario). Estas puertas son la clave para entender como la celda decide que información es necesaria en cada instante.

4.2.5.2. GRU

Una variación de las celdas LSTM son las celdas GRU, *gated recurrent unit*, introducidas en [41] que consiste en una simplificación de la celda original ya que se reduce el número de puertas en la celda sin perder la capacidad de gestionar las dependencias a largoy corto plazo. La principal ventaja de esta arquitectura simplificada es que funciona mejor con menos datos y requiere un coste computacional menor, todo ello sin perder rendimiento. En la figura 4.3 se puede ver el esquema de esta celda.

4.3 | Preparación de los datos. Ventana deslizante

En primer lugar se toma la decisión de realizar una división *train*, *validation* y *test* por una mera comodidad al rebajar los costes computacionales de realizar remuestreos u otras técnicas de *cross-validation* conocidas. En la figura 4.4 se puede observar nuestra división correspondiente a un (70,15,15) % respectivamente.

En concreto, nuestro trabajo aquí corresponde con un problema de aprendizaje supervisado en el que usaremos los valores de cada serie en los tiempos $t-1, t-$

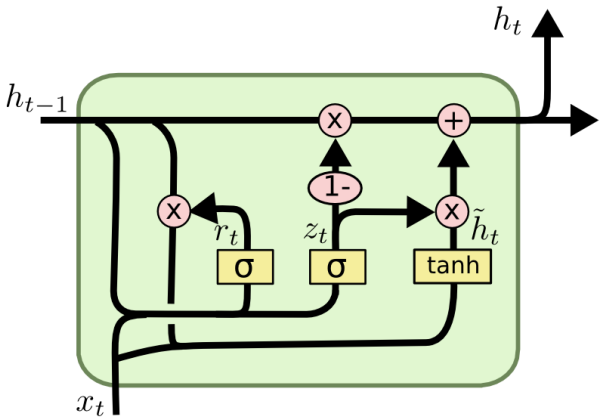


Figura 4.3: Representación de la estructura interna de una celda GRU. Fuente: [37].

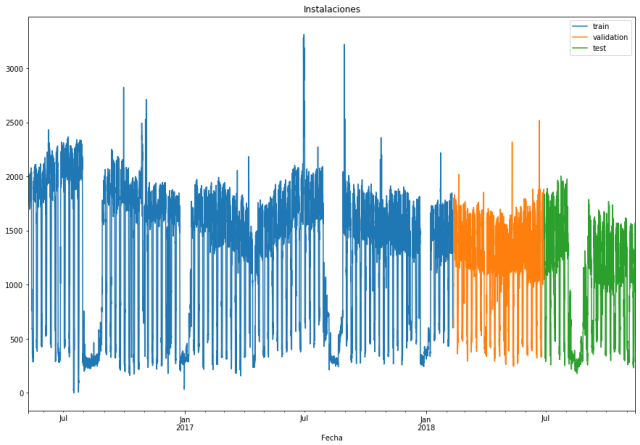


Figura 4.4: Representación de la división *train*, *validation* y *test* en la variable Instalaciones.

$2, ..., t-n$ para predecir el valor de t para la serie de la variable de Instalaciones. En concreto se usarán $t-1, t-2, ..., t-8$ para predecir el valor de t. Es decir, tendríamos un problema de series temporales multivariadas resuelto con una ventana temporal deslizante para predecir el paso siguiente. Este funcionamiento se ejemplifica en la figura 4.5.

A modo de homogeneización respecto a los modelos clásicos usados mantenemos el carácter predictivo en el paso siguiente a pesar de que estos modelos son más flexibles a la hora de modificar que paso quieres predecir solo cambiando los *outputs* seleccionados para el modelo [42].

Una vez se ha definido la ventana deslizante que se va a usar, en concreto de 24 pasos hacia atrás y uno hacia delante para predecir realizamos una ingeniería de variables con el fin de parametrizar la variable temporal. En concreto se han tomado las siguientes variables.

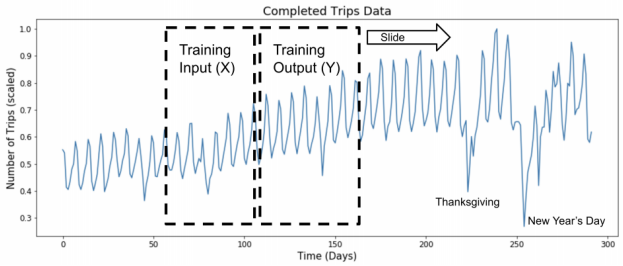


Figura 4.5: Ejemplo de representación de ventana temporal deslizante en un problema de series temporales para predicción. La imagen fue extraída de [43]. En nuestro caso de estudio la ventana corresponde con $\{t-8, t-7, ..., t-1\} \rightarrow t$

- Hora: Rango 0 a 23h.
- Día de la semana: Rango de 0 a 6, siendo 0 el lunes.
- Mes: Rango de 0 a 11, siendo 0 enero.
- Año: Rango 2016 a 2030.
- Semana del año: Rango de 0 a 53.
- Fin de Semana: Días de la semana 5 y 6.

Y estas se han transformado en *dummy variables*. En el análisis de regresión, una *dummy variable* toma la el valor 0 o 1 indica la ausencia o presencia de algún efecto categórico que se puede esperar que cambie el resultado [44]. En cualquier caso, en nuestro ejemplo lo hemos usado para poder usar estos algoritmos de forma más o menos eficiente con las series temporales pero esta técnica de crear variables ficticias es mucho más potente y general para éste u otros problemas.

4.3.0.1. Otras métricas de validación

Se ha seleccionado una métrica de validación adicional al *mse*. Este es el valor medio de los errores porcentuales simetrico, *sMAPE*. A pesar de que la metrica *mse* es perfectamente valido dentro de la industria a menudo se nos pide una forma de evaluar cual va a ser nuestro fallo en cada paso a dar de forma visual, en este sentido encaja perfectamente el *sMAPE* que se define a continuacion.

$$sMAPE = \frac{100\,\%}{n} \sum_{t=1}^{n} \frac{|F_t - A_t|}{(|A_t| + |F_t|)/2} \tag{4.3}$$

donde A_t representa el valor real y F_t es el valor predecido. El termino simétrico es una forma de solucionar el problema de la métrica mas clásica *MAPE* que beneficia a los pronósticos conservadores.

$$MAPE = \frac{100\,\%}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \tag{4.4}$$

4.3.1. Resultados

En la Tabla 2.1 se presentan los resultados obtenidos tras la aplicación de la pequenya muestra de algoritmos escogidos. En cuanto a las configuraciones tomadas en los algoritmos que proceden se comentaran a continuación.

Para la SVR lineal se ha usado con unos parámetros de C=0.1 y ϵ=0.01. El perceptron multicapa ha sido configurado con 5 capas de 1024 neuronas cada una con funciones de activación *relu* e intercaladas de *dropout* de 20 % como regularización. El tamaño del *batch* es 128, el optimizador escogido es *Adam* y la función de error *mse*. En cuanto a las redes GRU se ha optado por una configuración de una capa con celdas GRU de 300 unidades unida a una capa densa de una sola salida, intercalándose un *dropout* de 20 % . El tamanyo del *batch* es 64, el optimizador escogido es *Adam* y la funcion de error *mse*. En cuanto a las redes LSTM se ha optado por una configuración de una capa con celdas LSTM de 100 unidades unida a una capa densa de una sola salida, intercalándose un *dropout* de 20 % . El tamanyo del *batch* es 256, el optimizador escogido es *Adam* y la funcion de error *mse*. En general, la elección de hiperparametros en este estudio no ha seguido una estrategia exhaustiva debido a los costes computacionales y el delicado tiempo empleado en el desarrollo del conjunto.

Model	MSE	sMAPE
Persistencia.	13830.63	8.00 %
Regresión lineal.	**9212.63**	8.53 %
SVR lineal.	9497.55	**7.45** %
MLP.	41116.66	16.61 %
GRU.	13363.20	9.44 %
LSTM.	11132.31	9.01 %
SARIMA.	12354.21	9.67 %

Tabla 4.1: Resultados de los distintos modelos evaluados en base a las dos métricas elegidas.

5 | Conclusión

En primer lugar, me gustaría recordar que lo expuesto en este trabajo se trata de una primera aproximación al estudio de predicción sobre estas variables. Debido a esto en cada paso dado se ha sido plenamente consciente de las posibles mejoras y lineas a seguir ya que se debe conseguir un planteamiento adecuado que asegure unas líneas de trabajo futuras con altas expectativas.

En cuanto al apartado de preprocesado el camino seguido no ha resultado de gran innovación pero si ha permitido empaparnos de metodologías más útiles para tales propósitos así como tener la ligera intuición de una linea de trabajo en base a la detección de anomalías en series de tiempo (*outliers*) con el uso de redes neuronales recurrentes como se puede ver en [45].

Entrando ya en el punto de la modelización tenemos, al menos, tres lineas abiertas de trabajos futuros. En primer lugar el uso de modelos más complejos como pueden ser los modelos VARMA que son similares a los modelos ARMA pero introduciendo la posibilidad de ser multivariados. Esto conlleva varios problemas ya que la mayoría de las series descritas no son normales debido a la existencia de una componente en la distribución binomial (apagado/encendido) y éstos modelos funcionan adecuadamente para distribuciones de datos normales. Para lograr este enfoque podríamos dividir el modelado en dos problemas uno para clasificación encendido/apagado y el otro de regresión.

Por otro lado, se habían planteado dos líneas de predicción más. La primera dentro del mundo de las SVM que se ha comprobado que con una aproximación muy sencilla de usar kernels lineales ya aportan un rendimiento considerable al modelo. Una línea sería seguir con esta aproximación y pasar a usar KRR y, más concretamente, filtros kernel adaptativos y escalables [46]. Otra idea en este sentido es pasar a usar procesos gaussianos, siempre y cuando podamos asumir la distribución de datos normales como pasaba en el caso anterior. Los usaríamos en su versión dispersa debido a su bajo rendimiento computacional y, además, con una metodología online para modelar las series temporales [47, 48, 49].

En cuanto a la línea abierta con las redes neuronales recurrentes se debería investigar sobre un arquitectura más compleja. Por ejemplo, hacer uso de capas de autoencoders inicial que recoja las dependencias entre *features* de forma automática [50]. Otra idea recurrente en el estado del arte es el uso de capas convolucionales para crear representaciones de *features* de forma automática [51].

Una vez exploradas las distintas líneas se puede trabajar en un enfoque de *ensemble* de estos modelos a la hora de integrar los posibles distintos rendimientos de cada modelo en distintas partes de la muestra [52].

Como último aporte, se debe tener en cuenta que este trabajo supone una solución preliminar a una problemática industrial. Entonces, uno de los últimos objetivos es disponer de un algoritmo adecuado escalable para poder desplegar en el entorno de producción.

Bibliografía

[1] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis. Forecasting and control.* Prentice-Hall, 1994.

[2] J. Hamilton, *Time series analysis.* Princeton University Press, 1994.

[3] J. P. Brockwell and R. A. David, *Introduction to time series and forecasting.* Springer-Verlag, 1996.

[4] D. Peña, *Análisis de series temporales.* Alianza editorial, 2005.

[5] G. Glass, V. Willson, and J. Gottman, *Design and analysis of time-series experiments.* Colorado Associated University Press, 1975.

[6] C. W. J. Granger and P. Newbold, *Forecasting economic time series.* Academic Press Orlando, 1986.

[7] L. Lixia, "Nonlinear test and forecasting of petroleum futures prices time series," *Energy Procedia*, vol. 5, pp. 754–758, 2011.

[8] P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction.," *Nature.*, vol. 525, no. 7567, p. 47, 2015.

[9] R. McNown and A. Rogers, "Forecasting cause-specific mortality using time series methods.," *International Journal of Forecasting.*, vol. 8, no. 3, pp. 413–432, 1992.

[10] I. Odinaka, P. H. Lai, A. D. Kaplan, J. A. O'Sullivan, E. J. Sirevaag, S. D. Kristjansson, A. K. Sheffield, and J. W. Rohrbaug, "Ecg biometrics: A robust short-time frequency analysis.," *IEEE International Workshop on Information Forensics and Security.*, pp. 1–6, 2010.

[11] F. W. Lo and L. Tsai, "Deep learning for detection of fetal ecg from multi-channel abdominal leads.," *Asia-Pacific signal and information processing association annual summit and conference.*, 2018.

[12] D. Baron, "Machine learning in astronomy: A practical overview," *School of Physics and Astronomy Tel-Aviv University*, 2019.

[13] D. George and E. A. Huerta, "Deep learning for real-time gravitational wave detection and parameter estimation: Results with advanced ligo data.," *Physics Letters,.*, vol. B, no. 778, pp. 64–70, 2018.

[14] C. Feng, T. Li, and D. Chana, "Multi-level anomaly detection in industrial controlsystems via package signatures and lstm networks," *47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2017.

[15] R. T. Olszewski, *Generalized feature extraction for structural pattern recognition in time-series data.* PhD thesis, Carnegie Mellon University Pittsburgh, PA, USA., 2001.

[16] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. Moreno, "Speaker diarization with lstm.," *ICASSP*, 2018.

[17] F. R. R. Chowdhury, Q. Wang, I. Moreno, and L. Wan, "Attention-based models for text-dependent speaker verification.," *ICASSP*, 2018.

[18] D. Arriba, *Tecnicas de mejora del rendimiento de los sistemas de diarizacion de locutores*. PhD thesis, Universidad del pais vasco., 2016. David Tavarez Arriba.

[19] D. B. Rubin, "Inference and missing data.," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976.

[20] R. J. Little and D. B. Rubin, *Statistical analysis with missing data.* John Wiley & Sons., 2002.

[21] G. Williams, R. Baxter, H. He, S. Hawkins, and L. Gu, "A comparative study of rnn for outlier detection in data mining.," *IEEE International Conference on Data Mining*, 2002.

[22] H. Liu, S. Shah, and W. Jiang, "On-line outlier detection and data cleaning.," *Computers and Chemical Engineering*, vol. 28, pp. 1635–1647, 2004.

[23] D. Hawkins, *Identification of Outliers.* Chapman and Hall, 1980.

[24] V. Barnett and T. Lewis, *Outliers in statistical data.* John Wiley & Sons., 1994.

[25] T. Ane, L. Ureche-Rangau, J. B. Gambet, and J. Bouverot, "Robust outlier detection for asia-pacific stock index returns.," *J.Int.Financ.Mark.*, vol. 18, pp. 326–343, 2008.

[26] J. W. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.

[27] P. Tans, "Co2 expressed as a mole fraction in dry air, micromol/mol, abbreviated as ppm.," 2019.

[28] W. Playfair, *The Commercial and Political Atlas and Statistical Breviary*. 1786.

[29] T. Akaike, "Seasonal adjustment by a bayesian modelling.," *Journal of Time Series Analysis.*, vol. 1, no. 1, pp. 1–13, 1980.

[30] M. R. Young, "Robust seasonal adjustment by bayesian modelling.," *Journal of Forecasting.*, vol. 15, pp. 355–367, 1996.

[31] M. Lin, H. C. Lucas, and G. Shmueli, "Too big to fail: Large samples and thep-value problem..," *Information Systems Research.*, pp. 1–12, 2013.

[32] A. C. Harvey and P. H. J. Todd, "Forecasting economic time series with structural and box-jenkins models: A case study.," *Journal of Business Economic Statistics.*, pp. 299–307, 1983.

[33] C. Chatfield, *The analysis of time series: An introduction*. Chapman Hall., 1989.

[34] S. Rusell and P. Norvig, *Artificial intelligence: a modern approach.* 1995.

[35] C. Cortes and V. Vapnik, "Support-vector networks.," *Machine Learning.*, vol. 20, pp. 273–297, 1995.

[36] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors.," *Nature.*, vol. 323, p. 533–536, 1986.

[37] C. Olah, "Understanding lstm networks.," 2015.

[38] T. Young, D. Hazarika, S. Poria, and E. Cambria., "Recent trends in deep learning based natural language processing.," *IEEE Computational Intelligence Magazine.*, vol. 13, pp. 55 – 75, 2018.

[39] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult.," *IEEE Trans Neural Netw.*, vol. 5, pp. 157 – 166, 1994.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory.," *Neural computation.*, vol. 9, pp. 1735 – 1780, 1997.

[41] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation.," *Association for Computational Linguistics.*, p. 1724–1734, 2014.

[42] D. Kline, *Neural Networks in Business Forecasting.*, ch. Methods for Multi-Step Time Series Forecasting with Neural Networks. IGI Global, 2004.

[43] N. Laptev, J. Yosinski, L. E. Li, and S. Smyl., "Time-series extreme event forecasting with neural networks at uber,"

[44] D. B. Suits, "Use of dummy variables in regression equations.," *Journal of the American Statistical Association.*, vol. 52, pp. 548–551, 1957.

[45] D. Shipmon, J. Gurevitch, P. M. Piselli, and S. Edwards, "Time series anomaly detection: Detection of anomalous drops with limited features and sparse examples in noisy periodic data," tech. rep., Google Inc., 2017.

[46] P. H. Cédric Richard, José C. M. Bermudez, "Online prediction of time series data with kernels.," *IEEE Transactions on Signal Processing, Institute of Electrical and Electronics Engineers.*, vol. 57, p. 1058–1067, 2009.

[47] A. Ranganathan, M.-H. Yang, and J. Ho., "Online sparse gaussian process regression and its applications.," *IEEE Transactions on Image Processing.*, vol. 20, pp. 391–404, 2010.

[48] N. Lawrenceand, M. Seegerand, and R. Herbrich, "Fast sparse gaussian process methods: the informative vector machine.," *NIPS'02 Proceedings of the 15th International Conference on Neural Information Processing Systems.*, vol. 20, pp. 625–632, 2002.

[49] L. Csató and M. Opper, "Fast sparse gaussian process methods: the informative vector machine.," *Neural Computation.*, vol. 14, pp. 641–668, 2002.

[50] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory.," *PLOS ONE*, vol. 12, 2017.

[51] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. WOO, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems 28* (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds.), pp. 802–810, Curran Associates, Inc., 2015.

[52] S. Sun, Y. Wei, and S. Wang, "Adaboost-lstm ensemble learning for financial time series forecasting," in *ICCS*, 2018.